Amendments To the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1   Claim 1 (canceled).

1   Claim 2 (currently amended): A nonvolatile storage system comprising:

2   a controller capable of receiving commands from a host; and

3   a nonvolatile memory storage coupled to said controller, said <u>nonvolatile memory</u>

4   <u>storage</u> organized into blocks, one or more blocks caused to be identified by a group of logical

5   block addresses corresponding to a predetermined group of sectors<u>, each sector for including a</u>

6   <u>sector of data and</u>, each block [including] <u>storing</u> two or more sectors <u>of data,</u>

7   wherein said controller, in response to receiving a command from said host to rewrite

8   one or more sectors of data that are stored in said one or more blocks, writes said <u>sectors</u>

9   <u>of</u> data for said one or more sectors of data to be rewritten to one or more new blocks

10   caused to be identified by said group of logical block addresses without moving [or] <u>and</u>

11   <u>without</u> copying the <u>sectors of</u> data in the sectors in said one or more blocks that the host

12   did not specify to be rewritten in the command.

1   Claim 3 (currently amended): A nonvolatile storage system comprising:

2   a host for sending commands;

3   a controller coupled to said host for receiving host commands; and

4   nonvolatile <u>m emory</u> s torage<u>,</u> c oupled t o s aid c ontroller<u>,</u> f or s toring [sector information]

5   <u>sectors of data into sectors, said nonvolatile memory storage</u> organized into blocks, one or

6   more blocks caused to be identified by a group of logical block addresses corresponding to

7   a predetermined group of sectors, and each block [including] <u>having</u> two or more sectors

8   <u>of data,</u>

9   wherein said controller receives a command from said host for writing updated one or

10   more [sector information] sectors of data into a location within the nonvolatile memory

11   storage defined by said one or more blocks having previously-written [sector information]

12   sectors of data, other than that being updated by the host command, and wherein said

13   controller writes said updated one or more [sector information] sectors of data into one or

14   more new blocks caused to be identified by said group of logical block addresses thereby

15   avoiding moving [or] and without copying all of the previously-written [sector

16   information] sectors of data every time the host sends a command.

1   Claim 4 (currently amended): A nonvolatile storage system as recited in claim 3

2   wherein the controller further receives additional commands from the host for further

3   writing, one or more times, [sector information] sectors of data without moving the

4   previously-written [sector information] sectors of data every time [sector information]

5   sectors of data is updated.

1   Claim 5 (currently amended): A nonvolatile storage system as recited in claim 3

2   wherein the previously-written [sector information] sectors of data is moved from the

3   particular block at a time later than when the controller writes said updated one or more

4   [sector information] sectors of data to said new block.

1   Claim 6 (currently amended): A nonvolatile storage system as recited in claim 5 wherein the

2   particular block is erased at a time later than when the previously-written [sector information]

3   sectors of data is moved from the particular block.

1   Claim 7 (currently amended): A nonvolatile storage system comprising:

2   a host for sending commands;

3   a controller coupled to said host for receiving host commands; and

4   nonvolatile memory storage coupled to said controller for storing a sector of data, said

5   nonvolatile memory storage [sector information] organized into blocks, each block having two

6   or more sectors for storing [sector information] sectors of data, one or more blocks caused to

7   be identified by a group of logical block addresses corresponding to a predetermined group of

8   sectors, and each block including two or more sectors of data,

9   wherein said controller receives a command from said host for writing updated one

10   or more, but not all, [sector information] sectors of data into a location within the

11   nonvolatile memory storage defined by said one or more blocks having previously-written

12   [sector information] sectors of data, other than that being updated by the host command,

13   and wherein said controller writes said updated one or more [sector information] sectors of

14   data to one or more new blocks caused to be identified by said group of logical block

15   addresses thereby avoiding moving [or] and copying all the previously-written [sector

16   information] sector of data every time the host sends a write command.

1   Claim 8 (currently amended): A nonvolatile storage system as recited in claim 7

2   wherein the controller further receives additional commands from the host for further

3   writing, one or more times, [sector information] one or more sectors of data without

4   moving the previously-written [sector information] sectors of data every time [sector

5   information] the one or more sectors of data is updated.

1   Claim 9 (currently amended): A nonvolatile storage system as recited in claim 7

2   wherein the previously-written [sector information] sectors of data is moved from the

3   particular block at a time later than when the controller writes said updated one or more

4   [sector information] sectors of data to said new block.

1   Claim 10 (currently amended): A nonvolatile storage system as recited in claim 9

2   wherein the particular block is erased at a time later than when the previously-written

3   [sector information] sectors of data is moved from the particular block.

1   Claim 11 (currently amended): A method of updating information in a nonvolatile

2   memory storage having a controller coupled to a host and the nonvolatile memory storage

3   comprising:

4   receiving a command from the host for updating one or more [sector information]

5   sectors of data into a location within the nonvolatile memory storage defined by a particular

6   block having previously-written [sector information] one or more sectors of data other than

7   that being updated by the host command, said particular block caused to be identified by a

8   group of logical block addresses, corresponding to a predetermined group of sectors, and

9   including two or more sectors of data, each sector for storing a sector of data;

10   selecting one or more new blocks within the nonvolatile storage; and

11   writing said updated one or more [sector information] sectors of data to said one or

12   more new blocks caused to be identified by said group of logical block addresses without

13   moving [or] and without copying the previously-written [sector information] sectors of data.

1   Claim 12 (currently amended): A method of updating information as recited in claim 11

2   further including the step of receiving [further] additional commands from the host for further

3   updating, one or more times, [sector information] sectors of data wherein the previously-

4   written [sector information] sectors of data is not moved every time [sector information] a

5   sector of data is updated.

1   Claim 13 (currently amended): A method of updating information as recited in claim 11

2   further including the step of moving the previously-written [sector information] sector of data

3   from the particular block at a time later than said writing step.

1   Claim 14 (previously presented): A method of updating information as recited in claim 13

2   further including erasing the particular block at a time later than said moving step.

1   Claim 15 (currently amended): A nonvolatile storage system comprising:

2   a controller capable of receiving commands from a host; and

3   a nonvolatile memory storage, coupled to said controller, said <u>nonvolatile memory</u>

4   <u>storage</u> organized into blocks, each block having two or more sectors, <u>each sector</u> for storing

5   [sector information] <u>a sector of data,</u>

6   wherein said controller, in response to receiving a first write command from the host to

7   rewrite a first [sector information] <u>sector of data</u> defined by one or more sectors of

8   [information] <u>data</u> that are stored in a particular block, said particular block caused to be

9   identified by a group of logical block addresses, corresponding to a predetermined group of

10   sectors, and including two or more sectors, writes said first sector [information] <u>of data</u> to one

11   or more new blocks, said one or more new blocks caused to be identified by said group of

12   logical block addresses, without moving [or] <u>and without</u> copying sector [information] <u>of data</u>

13   previously-stored in the sectors of said particular block and not specified by the host in the

14   command to be rewritten, said controller, in response to receiving a second write command

15   from the host to rewrite a second sector [information] <u>of data</u> defined by sector [information]

16   <u>of data</u> within the particular block that is other than the particular sector [information] <u>of data,</u>

17   rewrites the second sector [information] <u>of data</u> into the particular block without moving the

18   first sector [information] <u>of data</u> [and] thereby preventing moving <u>a</u> sector [information] <u>of</u>

19   <u>data</u> every time a write command is received from the host.

1   Claim 16 (new): A nonvolatile storage system comprising:

2   controller coupled to a host for receiving host commands; and

3   nonvolatile memory storage coupled to said controller and organized into blocks having one

4   or more sectors for storing sectors of data, one or more blocks identified by a predetermined

5   group of logical block addresses corresponding to a predetermined group of sectors having

6   stored therein previously-written sectors of data, said controller receiving a host command for

7   writing updated one or more, but not all, sectors of data associated with said predetermined

8   group of sectors into a location within the nonvolatile memory storage defined by said one or

9   more blocks, said controller writing said updated one or more sectors of data to one or more

10   new blocks caused to be identified by said group of logical block addresses and upon

11   receiving a subsequent command to write subsequent sectors of data associated with said

12   predetermined group of a sectors but that are other than the updated one or more sectors of

13   data, the controller writing said subsequent sectors of data to the one or more new blocks and

14   only moving the previously-written sectors of data if, the previously-written sectors of data are

15   not updated.

16

1   Claim 17 (new): A nonvolatile storage system comprising:

2   controller coupled to a host for receiving host commands; and

3   nonvolatile memory storage coupled to said controller and organized into blocks having

4   sectors for storing data, one or more blocks identified by a predetermined group of logical

5   block addresses corresponding to a predetermined group of sectors having stored therein

6   previously-written data, said controller receiving a host command for writing updated one or

7   more, but not all, data associated with said predetermined group of sectors into a location

8   within the nonvolatile memory storage defined by said one or more blocks, said controller

9   writing said updated one or more data to one or more new blocks caused to be identified by

10   said group of logical block addresses and upon receiving a subsequent command to write

11   subsequent data associated with said predetermined group of a sectors but that are other than

12   the updated one or more data, the controller writing said subsequent data to the one or more

13   new blocks and only moving the previously-written data if, the previously-written data are not

14   updated.